

IoT senzor pro vysoké teploty

IoT sensor for high temperatures

Jan Talavašek

Bakalářská práce

Vedoucí práce: Ing. Radovan Hájovský, Ph.D

Ostrava, 2021

Abstrakt

Tato bakalářská práce se zabývá návrhem a realizací senzoru pro měření vysokých teplot, který bude využívat platformy internetu věcí Sigfox, a následně návrhem aplikace pomocí, které budou naměřená data zpracována a vizualizována. V první části jsou teoreticky rozebrány IoT platformy Sigfox, LoRa, NB-IoT a IQRF, vždy spolu s modulem určeným pro komunikaci s danou platformou. Ve zbylé části práce je popsán praktický návrh senzoru spolu s řetězcem zpracování a vizualizace.

Klíčová slova:

IoT; Měření teploty; Termočlánek; Měření vysokých teplot; Sigfox, NB-IoT; LoRa; IQRF, Grafana; UART; I2C; Vizualizace; Databáze; Mikrokontrolér; Zpracování dat

Abstract

The bachelor's thesis deals with the design and implementation of a sensor for measuring high temperatures, which will use the platform of the Internet of Things Sigfox, and then the design of an application using which the measured data will be processed and visualized. In the first part, the IoT platforms Sigfox, LoRa, NB-IoT and IQRF are theoretically analyzed, always together with the module intended for communication with the given platform. The rest of the work describes the practical design of the sensor together with the chain of processing and visualization.

Keywords:

IoT; Temperature measurement; Thermocouple; High temperature measurement; Sigfox; NB-IoT; LoRa; IQRF; Grafana; UART; I2C; Visualization; Database; Microcontroller; Data processing

Poděkování

Chtěl bych poděkovat vedoucímu práce Ing. Radovanu Hájovskému, Ph.D. za ochotu a věnovaný čas. Dále pak Ing. Janu Veličkovi za pomoc při řešení problematiky s výrobou plošných spojů a odlaďování programu. V neposlední řadě Marku Šindelářovi za poskytnutí serveru, nezbytného pro chod této práce.

Obsah

Seznam zkratk a symbolů	6
Seznam obrázků	7
Seznam rovnic	8
Úvod	9
1 Seznámení se s IoT platformami	10
1.1 IoT	10
1.1.1. Historie konceptu IoT	10
1.1.2. Rozdělení	10
1.1.3. Architektura IoT	11
1.2 Sigfox	12
1.2.1. Bezdrátový přenos	12
1.2.2. Architektura sítě	12
1.2.3. Přenosový protokol	13
1.2.4. Rozbor vysílacího modulu Sigfox Node	13
1.3 LoRa	14
1.3.1. Bezdrátový přenos	14
1.3.2. Architektura sítě	15
1.3.3. Přenosový protokol LoRaWAN	15
1.3.4. Rozbor vysílacího modulu RN2483A	15
1.4 NB IoT	16
1.4.1. Bezdrátový přenos	16
1.4.2. Architektura sítě	17
1.4.3. Přenosový protokol	18
1.4.4. Rozbor vysílacího modulu BC95	18
1.5 IQRF	19
1.5.1. Architektura sítě	19
1.5.2. Přenosový protokol DPA	20
1.5.3. Rozbor vysílacího modulu TR-72D	20

2	Návrh a realizace bezdrátového senzoru na bázi termočlánku a technologie Sigfox.	22
2.1	Návrh hardwarové části	22
2.1.1.	MCP9600	22
2.1.2.	ATmega328p	23
2.1.3.	Napájecí modul	23
2.2	Návrh softwarové části.....	24
2.2.1.	Sběrnice I2C.....	25
2.2.2.	Sběrnice UART	25
2.2.3.	Program a programovací jazyk.....	25
2.2.4.	Měření napětí baterie	25
2.2.5.	Vyčtení dat z MCP9600	26
2.2.6.	Zpracování dat.....	27
2.2.7.	Uspání MCU.....	27
2.3	Popis finální podoby navrhnutého IoT senzoru teploty	28
2.4	Měření na hotovém senzoru	29
3	Návrh softwarové aplikace pro zpracování a vizualizaci naměřených dat ..	30
3.1	Přeposlání dat ze Sigfox cloudu.....	30
3.2	Příjem dat ze Sigfox cloudu	31
3.3	Uložení dat	32
3.4	Vizualizace	33
4	Závěr	34
	Literatura	35
	Seznam příloh	37

Seznam zkratk a symbolů

AD	Analogově digitální
IoT	Internet of Things (Internet věcí)
ISM	Industrial, scientific, medical (volné pásmo pro rádiové vysílání)
ISP	In system programing
LAMP	Linux, Apache, MariaDB, PHP
MCU	Mikrokontroler
OS	Operační systém
SF	Spreading factor
UNB	Ultra narrow band

Seznam obrázků

Obrázek 1 - Zjednodušený řetězec datového toku v IoT	11
Obrázek 2 - Topologie sítě Sigfox	12
Obrázek 3 - Průběh odeslané zprávy [9]	13
Obrázek 4 - Modul Sigfox node [11].....	13
Obrázek 5 - Topologie sítě LoRa	15
Obrázek 6 - Modul RN2483A [15]	16
Obrázek 7 - Znázornění použití metody Standalone [17].....	16
Obrázek 8 - Znázornění použití metody In-band [17].	17
Obrázek 9 - Znázornění použití metody Guard-band [17].	17
Obrázek 10 - Topologie sítě NB-IoT	17
Obrázek 11 - Modul BC95-G [18]	18
Obrázek 12 - Topologie sítě IQRF	20
Obrázek 13 - Modul TR-72D [9].....	20
Obrázek 14 - Blokové schéma senzoru.....	22
Obrázek 15 - Schéma zapojení napájecího modulu	24
Obrázek 16 - Blokový diagram programu.....	24
Obrázek 17 - Formát zprávy předané modulu Sigfox.....	27
Obrázek 18 - Sestavený IoT senzor umístěný v krabici.....	28
Obrázek 19 - Měření spotřeby senzoru.....	29
Obrázek 20 - Relativní chyba senzoru	29
Obrázek 21 - Blokový digram zpracování dat	30
Obrázek 22 - Aktuální nastavení callbacku pro přeposílání dat na server	30
Obrázek 23 - Tabulka pro nastavení custom payload config	31
Obrázek 24 - Ukázka struktury databáze v příkazovém řádku	32
Obrázek 25 - Vizualizace dat ze senzoru v prostředí Grafana	33

Seznam rovnic

Rovnice 1 Výpočet napětí baterie z hodnoty vrácené AD převodníkem.....	26
Rovnice 2 Převod obdržených dat zpět na teplotu termočlánku	26

Úvod

Pojem IoT se stává trendem dnešní doby, ačkoliv se nejedná o novou myšlenku. Jeho rychle rostoucí rozvoj koresponduje s rozvojem technologií a možností přenášet a zpracovávat data rychleji a efektivněji. Dnes se IoT dotklo prakticky každé oblasti, v průmyslu jako průmysl 4.0, který umožňuje snížit prostředky na výrobu díky vyšší efektivnosti kterou zavádí. V domácnosti pak například jako řada zařízení označená obvykle názvem „smart“, díky kterým lze všední procesy automatizovat a navýšit tak komfort v domácím prostředí. Ačkoliv se jednotlivá IoT zařízení mohou značně lišit, tak jednu věc mají společnou, a to schopnost přenášet data do internetové sítě. Dle oblasti nasazení a jejich účelu k tomu pak využívají rozdílných technologií a infrastruktur které zprostředkovávají tuto komunikaci.

Jednou z nich je přenosová technologie Sigfox, která je navržena specificky pro IoT. Přesněji pro IoT zařízení, které přenášejí pouze malé množství dat nízkou rychlostí. Její využití je zpoplatněno licencí, avšak díky existující infrastruktuře a širokému pokrytí umožňuje velice rychlý vývoj výrobku.

Cílem této práce je navrhnout a prakticky sestavit senzor využívající technologie Sigfox, který bude schopen měřit vysoké teploty a následně vhodně zpracovat a vizualizovat naměřená data. Takovýto senzor pak najde své uplatnění při monitorování dlouhodobých vývojů teploty v místech, ke kterým je špatný přístup nebo mohou být pro člověka nebezpečné a je vhodné minimalizovat nutnost jejich návštěvy.

Práce je rozdělena do čtyř kapitol. V první z nich je vysvětlen koncept IoT a následně je proveden rozbor nejčastěji používaných platform pro realizaci IoT zařízení, u každé z nich jsou popsány její specifikace, přednosti i negativa v porovnání s ostatními, a proveden rozbor jednoho z modulů, který umožňuje komunikaci s danou platformou.

Následující kapitola je zaměřena na návrh hardwaru a softwaru senzoru. Při volbě prvků obvodu a návrhu programu pro mikrokontrolér je kladen důraz na co nejnížší spotřebu, tak aby senzor pracoval co nejdéle bez nutnosti výměny baterie. V úvodní části jsou popsány komponenty, které jsou využité pro realizaci. Práce pak pokračuje popisem navrženého programu, finální podobou senzoru, měřením přesnosti a spotřeby.

Posledním prvkem návrhu je softwarová aplikace pro zpracování a vizualizaci dat. Zde je postupně rozebrán celý navrhnutý řetězec, pomocí kterého jsou data přeposílána z cloudu společnosti Sigfox, zpracovávána, následně uložena do databáze a graficky zobrazena ve volitelném časovém rozmezí. V závěru, jsou pak zhodnoceny výsledky práce a možné budoucí vylepšení.

1 Seznámení se s IoT platformami

1.1 IoT

Internet věcí je novodobý termín označující síť fyzických objektů – zařízení, nástroje, auta, budovy, které jsou vybaveny technologiemi potřebnými pro sběr informací a připojeny k internetové síti. To umožňuje sesbíraná data hromadit, zpracovávat, přeposílat a zpětně samotné objekty řídit a ovládat. Díky této možnosti jsme schopni vytvářet a regulovat objekty mnohem efektivněji a přesněji.

1.1.1. Historie konceptu IoT

Název Internet of things byl poprvé použit Britským technologem Kevinem Ashtonem v roce 1999, který pracoval na zlepšení dodavatelského řetězce s využitím RFID tagů pro firmu P&G. Fráze Internet of Things pak tvořila titulní stránku jeho prezentace, a použil jí k popisu systému kde je internet propojen s fyzickým světem skrze senzory.

První zařízení schopné odesílat svá data skrze internet vzniklo již v roce 1982 na univerzitě Carnegieho–Mellonových. Jednalo se o upravený automat na nápoje, který sdílel informace o počtu zbývajících nápojů a jejich teplotě.

1.1.2. Rozdělení

Uživatelský internet věcí

Zde spadají především zařízení s označením „Smart home“ jako například chytré žárovky, pračky a kamery. Hlavním účelem těchto zařízení je zjednodušit úkony každodenního života. Vzájemným propojením pak lze určité procesy částečně nebo plně automatizovat. V neposlední řadě lze s využitím senzorů realizovat zabezpečení domácnosti.

Průmyslový internet věcí

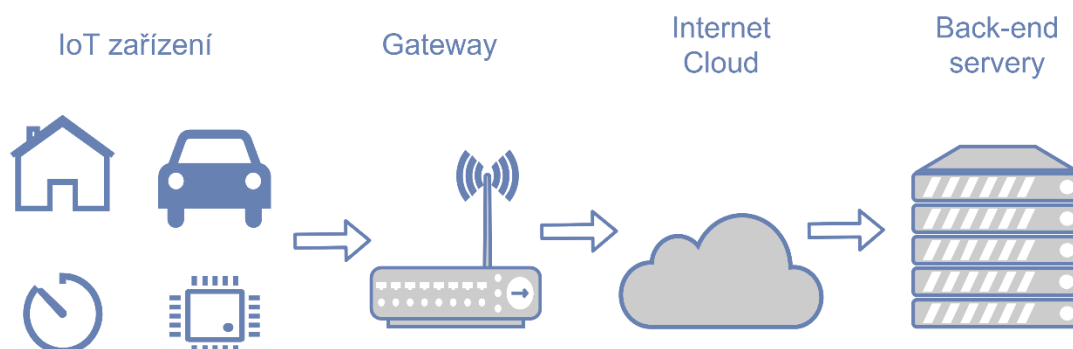
Propojením výrobních zařízení, vzniká systém, který lze snadno monitorovat. Za pomoci nasbíraných dat lze předcházet poruchám a jejich výměnou mezi zařízeními obvykle i celý proces zefektivnit. Jsou zde kladeny větší nároky na bezpečnost a robustnost.

Podnikový internet věcí

V nejjednodušší formě se jedná o implementaci IoT zařízení na pracovištích jako jsou tiskárny, docházkové systémy, kamery a RFID čtečky. V pokročilejší fázi můžou shromážděná data sloužit k rozhodování a opět zefektivňování procesů.

1.1.3. Architektura IoT

Propojení internetu s fyzickým světem v sobě zahrnuje mnoho faktorů a konkrétní řešení je vždy individuální pro danou aplikaci. Obecně lze však celý řetězec rozdělit do čtyř částí.



Obrázek 1 - Zjednodušený řetězec datového toku v IoT

IoT zařízení

V této vrstvě se nachází předměty schopné snímat a zaznamenávat parametry svého okolí, následně pak data předávat. Prakticky se jedná o hardware, který je obvykle specifikovaný ke konkrétnímu účelu.

Gateway

Gateway slouží jako brána mezi internetem a IoT zařízením. Jelikož přenos dat ze zařízení je vždy specifikován k dané aplikaci, tak je nutné realizovat zařízení které signál v dané formě zachytí a převede ho do formy ve které ho lze přenést skrze internet.

Internet

Internetová síť slouží v řetězci ke globálnímu propojení se servery, které se mohou nacházet kdekoli na světě, díky tomu je následně zajištěn i přístup k datům odkudkoliv.

Back-end servery

Poslední vrstva tedy back-end servery, obdržená data zpracovávají, ukládají do databází a umožňují s daty manipulovat.

1.2 Sigfox

Sigfox je Francouzská společnost, která vytvořila a zpravuje stejnojmennou technologii. Jedná se o komunikační síť, která umožňuje bezdrátový přenos malého objemu dat na dlouhé vzdálenosti, při nízké spotřebě energie. Společnost uvádí dosah až 40 km na volném prostranství a průměrně 10 km v zastavěných oblastech [12]. Dosah pak lze navýšit s použitím opakovačů.

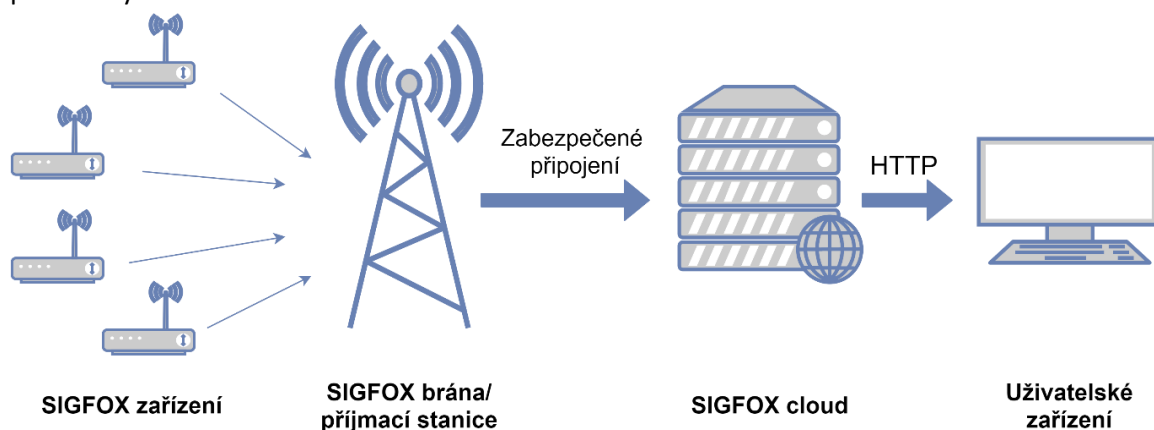
Sigfox poskytuje kompletní systém pro přenos včetně hardwaru, softwaru, protokolů, zpracování a cloudů. Použití je zpoplatněno licenci, její cena se odvíjí od konkrétního kontraktu se společností. Využití existující sítě však značně zkracuje dobu vývoje zařízení.

1.2.1. Bezdrátový přenos

Bezdrátový přenos probíhá v bezlicenčním pásmu ISM v Evropě na frekvenci 868Mhz. Přenos využívá ultra narrow band komunikace. Jedná se o technologii, která umožňuje velice efektivní komunikaci na velkou vzdálenost při zachování nízkého vysílacího výkonu, v případě Sigfox 25mW [12]. Vyslaný signál má pak podobu pulzu šířky 100 Hz, ten lze velice dobře demodulovat i když se jeho úroveň blíží hladině okolního šumu. Přenosová rychlost se pak pohybuje od 100 do 600 bitů/s v závislosti na použité frekvenci.

1.2.2. Architektura sítě

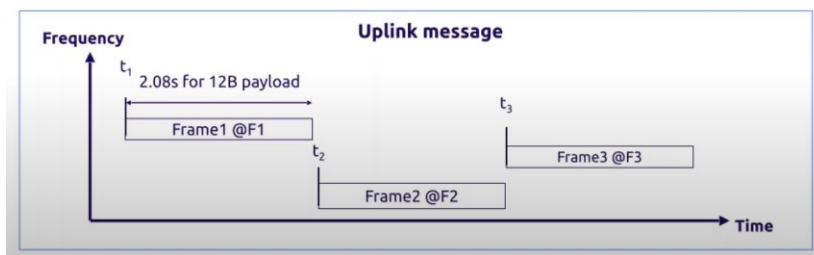
Sigfox pracuje s hvězdicovou topologií. Ta je tvořena sítí pozemních přijímacích stanic, které zachycují zprávy odeslané jednotlivými zařízeními. Pokud stanice zprávu zachytí, dojde k jejímu přeposlání do Sigfox cloudu. Pro přenos dat na Cloud využívá Sigfox lokálně dostupných operátorů, se kterými spolupracuje. V České republice je to společnost Simplecell, použitím jejich stávající infrastruktury je přenos uskutečněn. Na cloudu proběhne zpracování přijaté zprávy a její opětovné přeposlání do zákazníkem specifikovaného zařízení. Na Obrázek 2 je vyobrazen zjednodušený přenosový řetězec.



Obrázek 2 - Topologie sítě Sigfox

1.2.3. Přenosový protokol

Zpráva je přenášena ve formě paketu o maximální velikosti 12 bajtů a jejich množství je omezeno na 140 zpráv denně. Sigfox při odeslání dat využívá náhodného přístupu, vysílač není spárován s přijímačem a nedochází k žádné synchronizaci. Příkaz k odeslání dat je generován samotným vysílačem. Při odesílání je daný paket vyslán 3x po sobě na různých, avšak blízkých frekvencích, aby se snížila šance, že dojde ke kolizi dat. Při přenosu nedochází k žádnému potvrzení přijetí.



Obrázek 3 - Průběh odeslané zprávy [9]

Sigfox umožňuje přenášet data i zpět do vysílače tedy downlink, vysílač však musí o data zažádat. Poté se na 20 vteřin převede do režimu spánku a po probuzení čeká 25 vteřin na zažádanou zprávu. Pokud jí neobdrží přejde zpět do obvyklého režimu. Tato funkce je omezena na 4 zprávy denně o pevné velikosti 8 bajtů.

1.2.4. Rozbor vysílacího modulu Sigfox Node

Jedná se o modul společnosti Sigfox, který umožňuje přenos dat se stávající infrastrukturou přijímačů. Modul má podobu desky plošného spoje, která je osazena komponenty, pomocí kterých



Obrázek 4 - Modul Sigfox node [11]

jsou data modulována na nosnou vlnu o frekvenci 868Mhz a mikroprocesorem, který slouží jako řídicí člen. MCU je z výroby vybaveno systémem Sigfox a komunikace s ním je zajištěna skrze sběrnici UART, která je z desky vyvedena spolu s kontakty pro napájení. Požadavky jsou přes sběrnici zasílány ve formě AT příkazů. Jedná se o pevnou sadu příkazů, na které je schopen reagovat, jejich seznam je k dispozici v dokumentaci. Modul je dále osazen konektorem U.FL který slouží pro připojení externí antény. Většina komponentů je pak obalena plechovou schránkou, která slouží pro odrušení.

Pro uvedení modulu do provozu je nejdříve nutné zařízení aktivovat. To lze provést přiřazením zařízení k účtu registrovaném na stránkách backend.lpwan.cz na kterých je nutno zadat ID zařízení spolu s jeho PAC identifikátorem. V tu chvíli je na Sigfox zaslán požadavek na registraci zařízení spolu s informacemi o zakoupené licenci. Následně je zařízení připraveno komunikovat.

1.3 LoRa

LoRa je technologie přivedená na trh společností Semtech. Tato bezdrátová komunikační technologie umožňuje přenášet data na velkou vzdálenost s nízkou energetickou náročností. Dosah komunikace je závislý na nastavení vysílače. LoRa disponuje možností upřednostnit přenosovou rychlost před dosahem a naopak, tato vlastnost je popsána v části 1.3.1. Průměrně se však pohybuje okolo 40 km na volném prostranství a kolem 10 km v zastavěných oblastech.

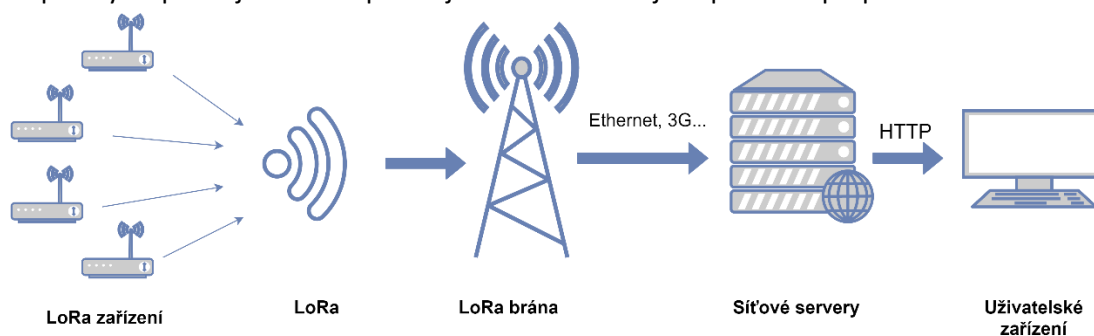
LoRa na rozdíl od Sigfoxu neposkytuje hotovou infrastrukturu. Jedná se pouze o specifikaci bezdrátové komunikace. Pro použití je nutné buďto vybudovat síť přijímačů a bran na vlastní náklady, případně využít existujících, které jsou zpoplatněny. Poslední možnost je využít volně přístupných přijímačů a bran které jsou vytvářeny komunitou po celém světě. Dalším rozdílem je že hardware pro komunikaci je dodáván různými výrobci.

1.3.1. Bezdrátový přenos

Bezdrátová komunikace probíhá stejně jako u Sigfoxu v bezlicenčním ISM pásmu 868Mhz. Signál je vysílán pomocí rozprostřeného spektra a využívá chirp modulace. Komunikační rychlost je závislá na spreading factoru, ten upravuje rychlost modulace pro dosažení buďto větší přenosové vzdálenosti nebo vyšší komunikační rychlosti. Je volen automaticky na základě kvality přenosu a je rozdělen do 5-ti úrovní od SF7 do SF12, přičemž SF7 má nejvyšší přenosovou rychlost 50 000 bit/s avšak nejkratší dosah. Naopak SF12 překlene největší vzdálenosti ale pouze s rychlostí 300bit/s. Vysílací výkon zařízení je omezen na 25mW.

1.3.2. Architektura sítě

LoRa využívá hvězdicovou topologii. Příjímací stanice jsou tvořeny koncovými zařízeními a LoRa branami. Zpráva odeslaná zařízením je zachycena všemi stanicemi v okolí a směrována na cloud skrze internetové připojení. V případě že je zpráva zachycena více stanicemi, jsou v cloudu odstraněny duplikáty a zpráva je uložena pouze jednou. Z cloudu jsou pak data přeposílána do zařízení uživatele.



Obrázek 5 - Topologie sítě LoRa

1.3.3. Přenosový protokol LoRaWAN

LoRaWAN je zabezpečený protokol pro přenos dat. Zpráva je odeslána ve formě paketu a její velikost je omezena na 256 bajtů, přičemž komunikace je plně obousměrná. Zabezpečení je realizováno za použití několika technik, základním prvkem je šifrování pomocí 128bitové šifry, kde je využito dvou klíčů, jeden má provozovatel sítě a druhý koncový zákazník.

Protokol LoRaWAN rozděluje zařízení do tří tříd, podle způsobu jejich použití.

Třída A – Zařízení pracující v této třídě umožňuje přijímat data pouze v krátkém časovém okně, které následuje po každém odeslání dat. Po uzavření tohoto okna již není příjem možný. Tato třída je nejméně energeticky náročná a je vhodná převážně pro zařízení která jsou napájena z baterií.

Třída B – Zařízení pracující v této třídě pracuje stejně jako ve třídě A, ale je zde možnost naplánovat otevření okna pro příjem v určitý čas a na zvolenou dobu.

Třída C – Zařízení pracující v této třídě je schopno přijímat data kdykoliv, kromě okamžiků, kdy probíhá jejich odesílání. Tato třída je nejvíce energeticky náročná a je vhodná pro zařízení napájená ze sítě, například LoRaWAN brány.

1.3.4. Rozbor vysílacího modulu RN2483A

Modul dodávaný výrobcem Microchip umožňuje komunikaci s technologií LoRa. Modul má podobu desky plošného spoje, která je osazena řídicím mikroprocesorem, k němuž jsou připojeny periferie zajišťující RF komunikaci. Ty dokážou data modulovat na frekvenci 433Mhz nebo 868Mhz podle zvoleného nastavení, stejně jako rychlost komunikace, která se pohybuje v rozmezí 10 kbitů/s až 300kb/s. Dále disponuje 14 GPIO piny přičemž 13 z nich může sloužit jako analogový vstup a komunikační sběrnici UART skrze kterou přijímá příkazy ve formátu ASCII. Pro zajištění kvalitního přenosu je nutné modul osadit anténou pro kterou je z pouzdra vyveden pin.



Obrázek 6 - Modul RN2483A [15]

1.4 NB IoT

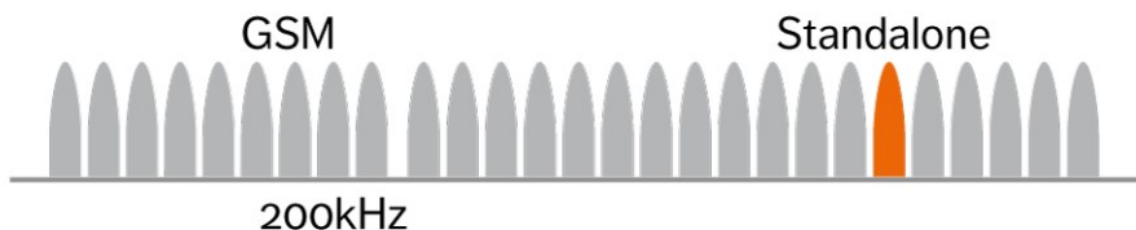
Narrow Band IoT je technologie vyvíjená projektem 3GPP tedy 3rd Generation Partnership Project. Tento termín označuje řadu normalizačních organizací, které vyvíjí protokoly pro mobilní telekomunikace. Technologie je vytvořena pro zasazení do stávající infrastruktury mobilních operátorů. Její dosah je tedy závislý na pokrytí sítě v dané lokaci. Ze všech technologií, které budou zmíněny dosahuje právě tato nejvyšší rychlosti přenosu a umožňuje přenést největší objem dat.

1.4.1. Bezdrátový přenos

Jak již bylo zmíněno technologie je zasazena do existující infrastruktury mobilních operátorů, využívá tedy na rozdíl od předešlých, licencovaného pásma 7Mhz až 900Mhz. Z tohoto důvodu je rovněž nutné, aby zařízení bylo vybaveno speciální SIM kartou, pomocí které může komunikaci navázat. Výhodou je však že pro implementaci této technologie stačí pouze softwarová úprava stávající infrastruktury.

NB-IoT rozlišuje tři způsoby zasazení standalone, in-band a guard-band.

Standalone – Tato metoda využívá nasazení NB IoT místo GSM nosiče, který pracuje v pásmech 900MHz až 1800MHz.



Obrázek 7 - Znázornění použití metody Standalone [17].

In-band – Následující metoda používá flexibilní části LTE nosiče, do kterého je vložen nosič NB-IoT.



Obrázek 8 - Znázornění použití metody In-band [17].

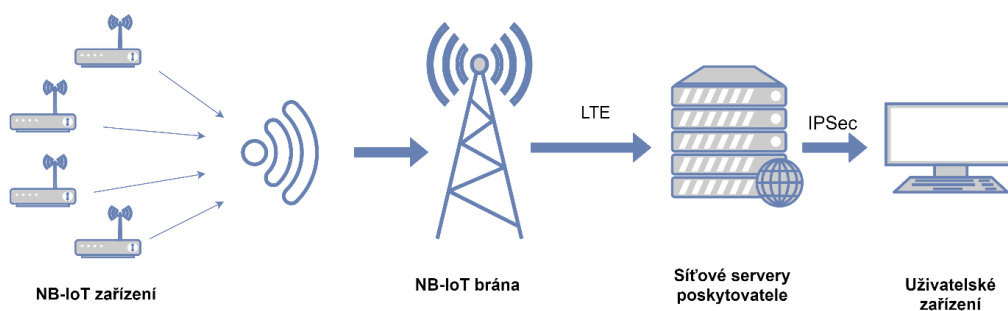
Guard band – Poslední metoda spočívá ve vyčlenění ochranného pásma v nosiči LTE, které je nahrazeno NB IoT nosičem.



Obrázek 9 - Znázornění použití metody Guard-band [17].

1.4.2. Architektura sítě

NB-IoT používá stávající hvězdicovou architekturu sítě LTE, která je upravena softwarovou aktualizací pro nasazení této technologie. V české republice zasadili NB-IoT do své sítě operátoři Vodafone a T-mobile. Data odeslána koncovým zařízení jsou zachycena branami pro NB-IoT, následně přenesena pomocí LTE protokolu na servery poskytovatele, kde proběhne jejich přeposlání do zařízení uživatele skrze zabezpečený IPSec tunel. Struktura je znázorněna na Obrázek 10.



Obrázek 10 - Topologie sítě NB-IoT

1.4.3. Přenosový protokol

Přenosový protokol umožňuje oboustrannou komunikaci, přičemž počet zpráv není omezen, maximální délka zprávy je 1280 bajtů při přenosové rychlosti 50kbps. Zabezpečení přenosu podléhá LTE šifrování.

1.4.4. Rozbor vysílacího modulu BC95

Jedná se vlastně o řadu modulů, které jsou vzájemně kompatibilní, odlišují se jen na základě frekvence B1/B3/B5/B8/B20/B28. V modulu je řídicí člen zastoupen procesorem ARM Cortex A7 ke kterému jsou připojeny potřebné periferie jako paměti a čip obstarávající RF komunikaci. Procesor je řízen pomocí AT příkazů po sběrnici UART, které jsou k dispozici dvě, přičemž druhá slouží pro debugging. Dále modul disponuje sběrnici pro připojení SIM karty. Ta je nutná pro autorizaci zařízení v síti. Zařízení je rovněž nutné osadit externí anténou.



Obrázek 11 - Modul BC95-G [18]

1.5 IQRF

IQRF je bezdrátová technologie patřící České firmě Microrisc. Od předchozích se značně liší, jelikož neumožňuje přenášet data na velkou vzdálenost, spojení lze navázat pouze v řádu desítek až stovek metrů v závislosti na prostředí. Přenos samotný je zaměřen na nízké objemy dat s nízkou rychlostí.

Společnost poskytuje kompletní systém jedné značky včetně hardware, software, protokolů a podpory.

1.5.1. Architektura sítě

Fyzická topologie sítě využívá mesh architektury, není tedy závislá na existující infrastruktuře. Sít' je tvořena jednotlivými zařízeními. Ty jsou schopny pracovat ve dvou režimech jako Node a Coordinátor

Node – Je prvek v síti mesh, který vykonává v základu dvě funkce. První z nich je sběr a odeslání dat, případně příjem a vykonání instrukce. Jeho druhou funkcí je přeposílání dat z ostatních nodů, funguje tedy jako repeater. Node lze konfigurovat tak aby nevykonával přeposílání, toho se využívá u okrajových zařízení sítě, která nemají od koho data obdržet. Nebo u zařízení napájených z baterií, deaktivací funkce je značně snížena spotřeba energie. Tento prvek je v síti zastoupen nejvíce

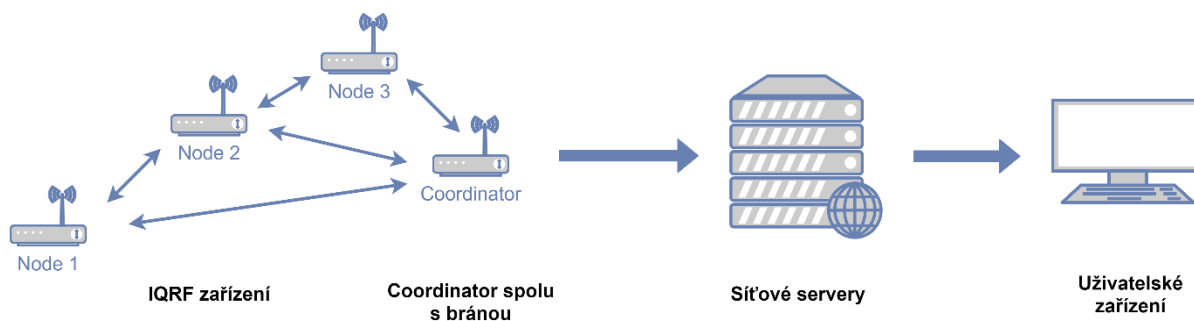
Coordinator – slouží k řízení celého procesu přenosu dat. Vysílá příkazy nodům a zároveň jsou do něj směrována veškerá data. Při použití základního protokolu lze na jednoho koordinátora připojit až 239 nodů.

Každé zařízení v síti je schopné pracovat jako node i jako koordinátor, jeho funkci je mu nutné přidělit při konstrukci sítě, při této proceduře je rovněž nutné přidělit nodům routovací číslo, to slouží pro zamezení kolize dat při přeposílání a je závislé na fyzickém umístění nodu v síti.

Přenos je pak skutečně pomocí patentovaného protokolu IQMESH, který využívá směrovaného zaplavení. V případě odeslání dat nodům vyšle Coordinátor příkaz, ten obdrží všichni Nodi v dosahu. Ti příkaz vykonají, přepošlou data dalším nodům v dosahu a koordinátorovy odešlou odezvu. Celý proces se opakuje, dokud nejsou „zaplaveni“ všechna Node zařízení. Data jsou odesílána v jednotlivých krocích podle přiděleného routovacího čísla, a to v pořadí od nejmenšího po největší.

Pro zaslání dat opačným směrem platí obdobný postup, data jsou přeposílána všemi Nody v dosahu a směrována ke koordinátorovi. Ten pak obdrží duplicitní data s časovým zpožděním. Přenos je tedy redundantní a je zde velice malé riziko ztráty dat.

Coordinátor pak s využitím brány, které bývá často součástí, odešle data skrze internet na uživatelem definovanou adresu. IQRF lze provozovat i v režimu bez koordinátorů, jedná se pak o peer to peer komunikaci, tato varianta však postrádá veškeré výhody spojené s IQMESH.



Obrázek 12 - Topologie sítě IQRF

1.5.2. Přenosový protokol DPA

Data jsou odesílána ve formě paketů o maximální velikosti 64 bajtů. Komunikace probíhá v bezlicenčním pásmu ISM na frekvenci 868MHz rychlostí 20kb/s. Čas nutný pro doručení zprávy je pak dán součinem času odesílání dat a počtem opakování (skoků) nutných pro doručení. Pro určení času odezvy platí stejný vztah, akorát je nutné uskutečnit dvojnásobný počet skoků.

1.5.3. Rozbor vysílacího modulu TR-72D

Jde o modul od společnosti IQRF. Je schopen komunikovat na několika frekvencích, podle regionu, ve kterém je nasazen, a to 868MHz 433Mhz a 916Mhz. Má podobu oboustranné desky plošného spoje, která je dodávána ve dvou variantách, s integrovanou anténou na plošném spoji nebo s variantou kde je anténa vyvedena na konektor U.FL a zároveň na pájecí plošku.

Modul je osazen řídicí jednotkou PIC16LF1938 který zprostředkovává komunikaci s využitím integrovaných RF periférií, dále disponuje teplotním čidlem, sběrnicemi UART,I2C,SPI šestící GPIO pinů, integrovaným AD převodníkem a dvojicí LED diod které signalizují provozní stav.



Obrázek 13 - Modul TR-72D [9]

Veškeré piny jsou pak vyvedeny ve spodní části desky tak, aby modul bylo možné zasunout do SIM patice. Modul vyžaduje napájení v rozsahu 3,1 až 5,3V a maximální výkon je 10mW.

Z výroby je MCU vybaveno systémem IQRF, který se skládá ze dvou vrstev. První z nich je operační systém společnosti a druhá je tvořena aplikací uživatele, ta je však operačnímu systému podřazena. Následné zavedení modulu do sítě lze provést buď manuální změnou napěťové úrovně na pinu k tomu definovaném, nebo vzdáleně kdy jsou kdy jsou potřebné informace vyčteny a zapsány do koordinátora.

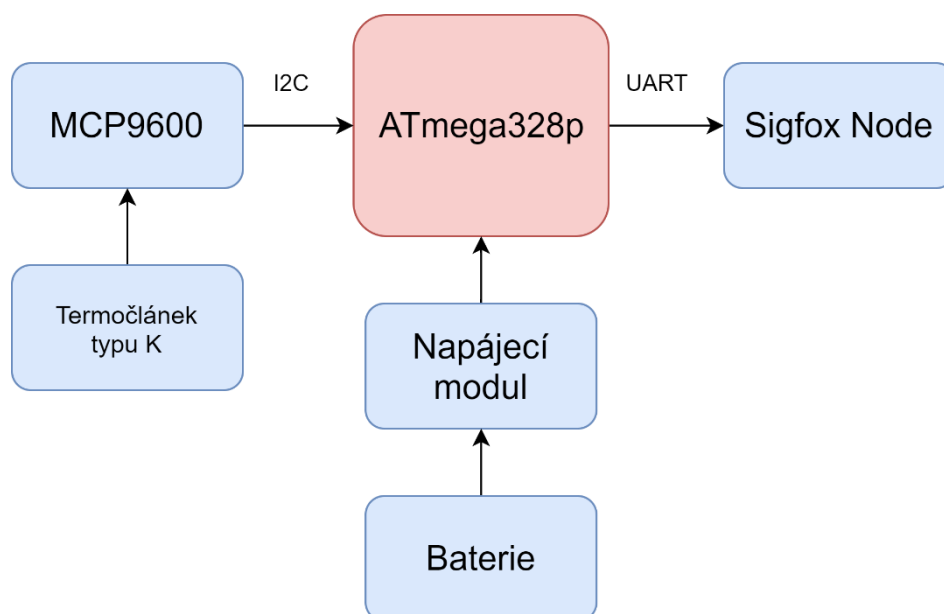
Pro programování modulu lze využít dvou možností, a to buď programováním přímo pod OS kdy je do druhé vrstvy vložena uživatelská aplikace v jazyce C, nebo s využitím protokolu DPA a uživatelského obslužného programu. Druhá zmíněná možnost jako jediná podporuje využití IQMESH.

2 Návrh a realizace bezdrátového senzoru na bázi termočlátku a technologie Sigfox.

V této části práce se budu zabývat návrhem senzoru. Návrh bude rozdělen na dvě hlavní části, a to hardwarová a softwarová, přičemž každá z nich bude sestávat z popisů jednotlivých částí uvedených v blokových diagramech, kromě modulu Sigfox Node, který je popsán v kapitole 1.2.4.

2.1 Návrh hardwarové části

Navrhovaný senzor bude schopen provádět měření vysokých teplot a tento údaj přenášet na internetovou síť za pomoci technologie Sigfox a modulu Sigfox Node. Zároveň bude energeticky úsporný, aby byl zajištěn jeho dlouhodobý provoz na baterii. Senzor bude jako snímací prvek využívat termočlánek. Jednotka pro zpracování a úpravu dat je tvořena mikroprocesorem ATmega328p, který bude měřenou teplotu vyčítat z paměti obvodu MCP9600.



Obrázek 14 - Blokové schéma senzoru

2.1.1. MCP9600

Obvod MCP9600 je integrovaný převodník termoelektrického napětí na stupně Celsia od výrobce Microchip. Podporuje širokou škálu standartních typů termočlávků. Pracuje v napájecím rozsahu 2,7 až 5,5 voltů a komunikace s ním probíhá po dvou vodičové sběrnici I2C. Standartně je dodáván v pouzdře MQFN.

Napětí generované termočlávkem je přiváděno na vstup skrze externí dolnoproustní RC filtr, ten slouží pro potlačení rušivých vlivů vedení. Pro digitalizaci signálu slouží v obvodu 18bitový AD převodník typu sigma delta, který využívá interní napěťové reference. Obvod disponuje vlastním teplotním čidlem okolní teploty a na základě jeho hodnoty provádí i kompenzaci studeného konce termočlátku.

Při kontinuálním měření dochází k zahřívání obvodu, z tohoto důvodu je ve středové části pouzdra umístěna ploška, kterou je nutné spojit s deskou plošného spoje tak, aby byl tepelně stabilní a nedocházelo k nesprávnému odečtu teploty.

Obvod umí pracovat ve třech režimech, kromě zmíněného kontinuálního měření disponuje také režimem spánku, ve kterém je jeho spotřeba téměř nulová a režimem bursts. V tomto režimu se obvod nachází ve stavu nízké spotřeby a očekává po sběrnici příkaz na základě kterého provede sérii měření, které následně zprůměruje. Výslednou hodnotu pak uloží do registrů, které zůstávají přístupné po celou dobu a přechází zpět do stavu nízké spotřeby.

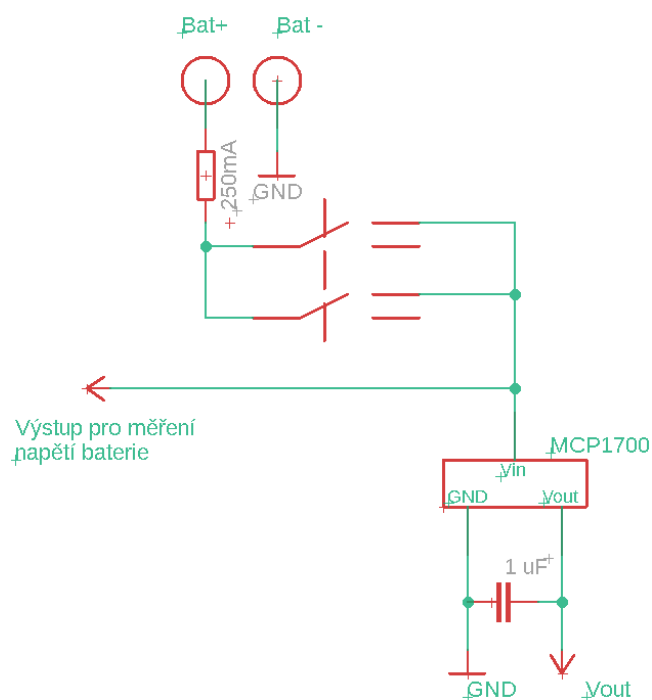
2.1.2. ATmega328p

Jde o 8bitový mikroprocesor s architekturou AVR od bývalé firmy Atmel, nyní nesoucí název Microchip, který je taktován interním krystalovým oscilátorem na frekvenci 8 MHz. Procesor je schopen pracovat i na vyšších taktech (až 20Mhz) avšak v této aplikaci to není potřeba. Nižší takt zajišťuje menší spotřebu procesoru, posouvá dolní hranici napájecího napětí a snižuje množství potřebných součástek. Procesor podporuje ISP, tedy možnost jeho programování bez nutnosti jej vyjmout ze zařízení. V této práci je ISP využito a potřebné piny jsou vyvedeny na desku plošného spoje v podobě pinové lišty. Dále disponuje potřebnými sběrnici I2C a UART skrze které probíhá komunikace s ostatními prvky senzoru.

2.1.3. Napájecí modul

Regulace napájecího napětí pro všechny prvky v obvodu je realizována přímo na desce senzoru. Jako zdroj slouží lithiová baterie (LI-SOCL₂) s nominálním napětím 3,6V v podobě C článku, která bude vložena do držáku a k desce připojena pomocí vodičů. Tyto baterie dosahují vysoké energetické hustoty a jejich kapacita se v tomto balení pohybuje průměrně okolo 8,5Ah. Baterie má rovněž vhodnou vybíjecí charakteristiku, kdy zůstává na své nominální hodnotě napětí téměř skrze celý vybíjecí cyklus [23].

Monitorování stavu baterie je zajištěno přivedením jejího napětí skrze napěťový dělič tvořený rezistory 300k a 100k na vstup 10bitového AD převodníku mikrokontroléru, kde je snímáno s využitím interního referenčního napětí 1,1V.

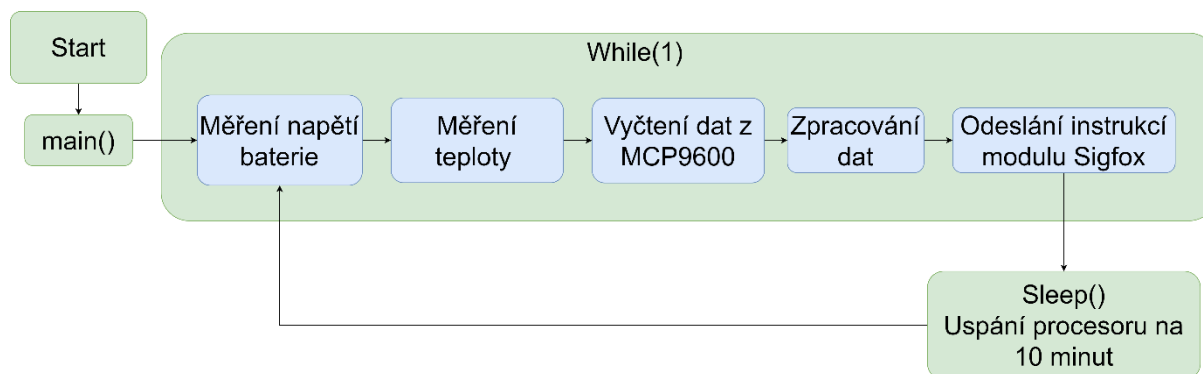


Obrázek 15 - Schéma zapojení napájecího modulu

Pro zajištění stabilního napětí v obvodu slouží regulátor MCP1700, jedná se o pevný lineární regulátor pracující s CMOS tranzistorem. Je dodáván ve velkém rozpětí výstupních napětí, přičemž zde bude použita varianta 3,3V. Obvod je schopen dodat proud až 250mA při poklesu na jeho přechodu 178mV. Dále je vybaven ochranou proti zkratu na výstupu a přehřátí. Výstup je osazen kondenzátorem 1uF pro stabilizaci špiček vzniklých náhlou změnou odběru. Na straně baterie je pak osazena pojistka spolu s vypínačem.

2.2 Návrh softwarové části

Pro funkčnost snímače bylo nutné vytvořit program kterým bude MCU, popsáný v kapitole 2.1.2 řízen. Před samotným popisem jednotlivých částí kódu a jazyku ve kterém je napsán, se budu věnovat stručnému popisu sběrnic po kterých probíhá komunikace.



Obrázek 16 - Blokový diagram programu

2.2.1. Sběrnice I2C

Jedná se dvouvodičovou multi-master, multi-slave sběrnici, která je v této práci použita pro přenos dat mezi MCP9600 a mikrokontrolérem. Komunikace po sběrnici je synchronní a pro synchronizaci slouží jeden ze dvou vodičů označený jako SCL tedy serial clock. Samotná data jsou pak přenášena vodičem SDA tedy serial data. Každé zařízení na sběrnici má svou unikátní adresu, na základě, které je voláno. Pro funkčnost komunikace je nutné připojit oba vodiče ke kladné napěťové hladině skrze vhodný rezistor. Sběrnice je konstruována pouze pro komunikaci na krátkou vzdálenost, v rámci desky plošného spoje.

2.2.2. Sběrnice UART

Asynchronní sériové rozhraní UART je využito pro komunikaci s modulem Sigfox Node. Jedná se opět o dvouvodičovou sběrnici, přičemž data jsou vysílána na pinu TX tedy transmit a přijímána pinem RX receive. Rychlost komunikace je nastavena podle modulu Sigfox na 9600 baudů za sekundu.

2.2.3. Program a programovací jazyk

Program je psán v jazyce C++. Jedná se o multiplatformní programovací jazyk, který je rozšířením staršího jazyka C, oproti němu přináší podporu několika programovacích stylů jako například objektově orientované programování. Díky své rozšířenosti je pro něj k dispozici široká podpora a velké množství knihovných prvků. Základní struktura kódu sestává z funkce main, ve které se nachází samotný kód. Před vstupem do této funkce jsou definovány globální proměnné a je provedeno zahrnutí potřebných knihoven.

2.2.4. Měření napětí baterie

Pro měření stavu baterie je nejprve nutné nastavit AD převodník. Ačkoliv je v MCU k dispozici 8 vstupů pro převod analogové hodnoty na digitální, tak disponuje pouze jedním 10bitovým AD převodníkem s postupnou aproximací, před kterým je zařazen multiplexor. V kódu je tedy nejprve multiplexor nastaven tak aby na převodník přivedl hodnotu z analogového pinu ADC1. Tento typ převodníku vyžaduje pro svou funkci hodinový signál v rozsahu 50kHz – 200Khz, ten je získán z hodinového signálu mikrokontroléru, který je následně snížen prescalerem na požadovaný rozsah. V této práci MCU pracuje na frekvenci 8Mhz a prescaler jsem nastavil na hodnotu 128. Výsledný hodinový signál má pak frekvenci 62,5Khz. Posledním úkonem je nastavení referenčního napětí, zde jsem využil interní napěťové reference 1,1V.

Po provedení nastavení je zahájeno měření přivedeného napětí a vrácená hodnota v rozsahu 0–1024 je uložena do proměnné. Jelikož je napětí baterie přivedeno skrze napěťový dělič je nutné hodnotu přepočítat. V první fázi je hodnota z rozsahu 0–1024 přepočtena na reálné napětí na vstupu převodníku, následně je za pomoci trojčlenky určeno napětí před děličem, tedy napětí baterie.

Rovnice 1 - Výpočet napětí baterie z hodnoty vrácené AD převodníkem

$x = \text{hodnota vrácená převodníkem}$

$V_{adc} = (1.1/1023) * x$

$V_{cc} = (V_{adc}/0.011)/100 * 4.38245$

Popis veličin

- V_{adc} – Napětí na vstupu AD převodníku
- V_{cc} – Napětí baterie

2.2.5. Vyčtení dat z MCP9600

Pro práci s obvodem jsem v programu vytvořil čtyři funkce a to MCPStart, MCPGetStatus, MCPMeasure, MCPStop, které jsou v tomto pořadí postupně volány a jejich jednotlivé funkce podrobně rozeberu níže.

První volanou funkcí je MCPStart, v ní je proveden zápis do konfiguračního registru zařízení, který slouží pro nastavení parametrů jako jsou jeho provozní režim a rozlišení AD převodníku teplého a studeného konce. Při zápisu nastavuji zařízení do normálního operačního režimu s nejvyšším možným rozlišením.

MCPGetStatus, poté co je obvod převeden do normálního operačního režimu, tak začne periodicky měřit teplotu termočlánku a následně data ukládá. Tato funkce kontroluje bit ve status registru zařízení, který označuje úspěšné dokončení nového měření a uložení dat do paměti. Následně vrací hodnotu tohoto bitu, tedy false pokud měření probíhá a true jakmile je dokončeno a data jsou k dispozici. Funkce je volána do té doby, dokud nevrátí hodnotu true a program může pokračovat.

MCPMeasure, jakmile jsou naměřená data k dispozici je touto funkcí provedeno jejich vyčtení. To probíhá ve dvou krocích, jelikož hodnota má velikost 16 bitů. Nejprve MCU zažádá o prvních 8 vyšších bitů a následně o zbylých 8 nižších bitů. Obě zprávy jsou následně spojeny a převedeny na teplotu s využitím vzorce obsaženého v dokumentaci. Po vyčtení teploty je provedeno nulování bitu ve status registru, který označuje dokončené měření, aby při příštím měření nedošlo k vyčtení původních dat. Funkce pak výslednou teplotu vrací ve formě double.

Rovnice 2 - Převod obdržených dat zpět na teplotu termočlánku

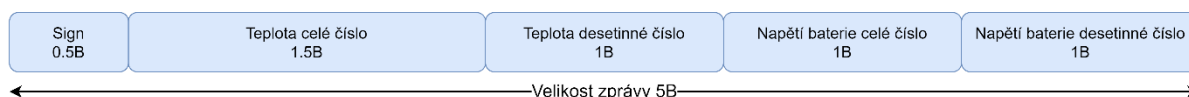
```
//Convert the temperature data
if ((UpperByte & 0x80) == 0x80){ //Temperature < 0°C
    Temperature = (UpperByte x 16 + LowerByte / 16) - 4096;
}else //Temperature ≥ 0°C
    Temperature = (UpperByte x 16 + LowerByte / 16);
```

MCPStop, pak převede zařízení do režimu shutdown, pro snížení spotřeby, opět zápisem do konfiguračního registru.

2.2.6. Zpracování dat

Modul Sigfox Node je schopen přijmout data pouze v přesně definovaných formátech. V této práci jsem se rozhodl předávat data v hexadecimální podobě. Před převodem do jiné soustavy je však nutné uložené hodnoty rozdělit na celou a desetinnou část a ty pak uložit do separátních proměnných. Pro oddělení celé části jsem hodnotu převedl na proměnou typu integer. Desetinná část je pak uložena jako zbytek po celočíselném dělení stem, přičemž před tím je hodnota stem vynásobena. U měřené teploty je před převodem určeno, zda je kladná či záporná s využitím podmínky a tato informace je uložena do proměnné sign.

Výsledkem tohoto zpracování je tedy pět proměnných, které jsou následně vloženy do pole typu string a předány Sigfox modulu po sběrnici UART, spolu s AT příkazem pro jejich odeslání.



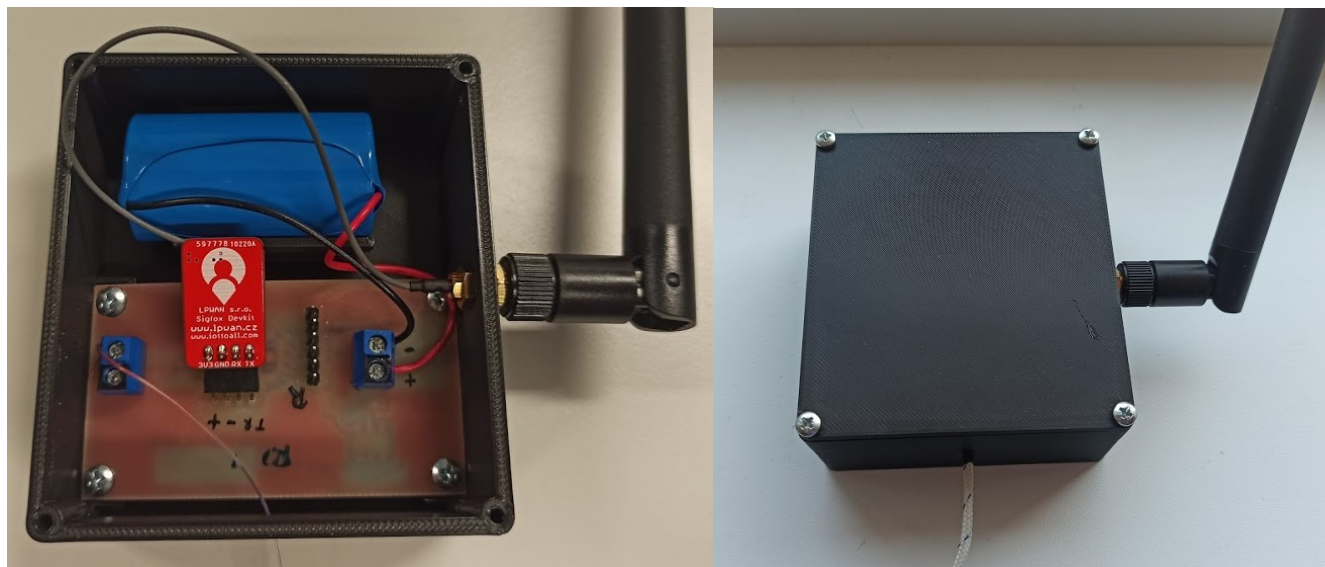
Obrázek 17 - Formát zprávy předané modulu Sigfox

2.2.7. Uspání MCU

Atmega328 disponuje možností přejít do několika módů režimu spánku, kdy jeho spotřeba klesá v závislosti na nastavení. V této práci využívám mód spánku s nejnižší spotřebou nazvaný PWR_DOWN. Při aktivaci tohoto módu je zastaven interní oscilátor procesoru a tím veškeré operace. Probuzení je pak možné pouze externím interruptem, skrze sériovou sběrnici nebo watchdogem. Pro zachování jednoduchosti obvodu jsem rozhodl procesor probouzet časovačem watchdogu který pracuje na vlastním RC oscilátoru. Jeho časování je nastaveno na maximální možnou hodnotu 8 sekund, to by však byla nedostatečně dlouhá doba, proto po probuzení procesor pouze inkrementuje proměnou ve funkci for a přechází zpět do režimu spánku. Cyklus se opakuje, dokud proměnná nedosáhne požadované hodnoty a program skáče na začátek. Hodnotu jsem určil experimentálně jelikož samotné probuzení, inkrementace a uspání zabírá určitý čas který je nutný přičíst k nastaveným osmi sekundám. Výsledkem experimentu je hodnota 73 která odpovídá odeslání zprávy každých deset minut a dvacet vteřin, tedy celkově 139 zpráv denně.

2.3 Popis finální podoby navrhnutého IoT senzoru teploty

Finální vzhled lze vidět na Obrázek 18. Senzor má podobu jedné desky plošného spoje, který je osazen šroubovacími svorkami pro připojení termočlánu a napájení. Dále pak patičí pro pinovou lištu, na kterou jsou vyvedeny čtyři kontakty pro připojení modulu Sigfox node. Posledním prvkem umožňující připojení je pinová lišta sestávající z šesti kontaktů, která slouží pro externí programátor mikrokontroléru. Kontakty lišty jsou záměrně seřazeny tak aby vedle sebe byly vyvedeny piny pro reset MCU a ground. Vložením propojky pak lze zastavit chod programu, pokud je to potřeba.



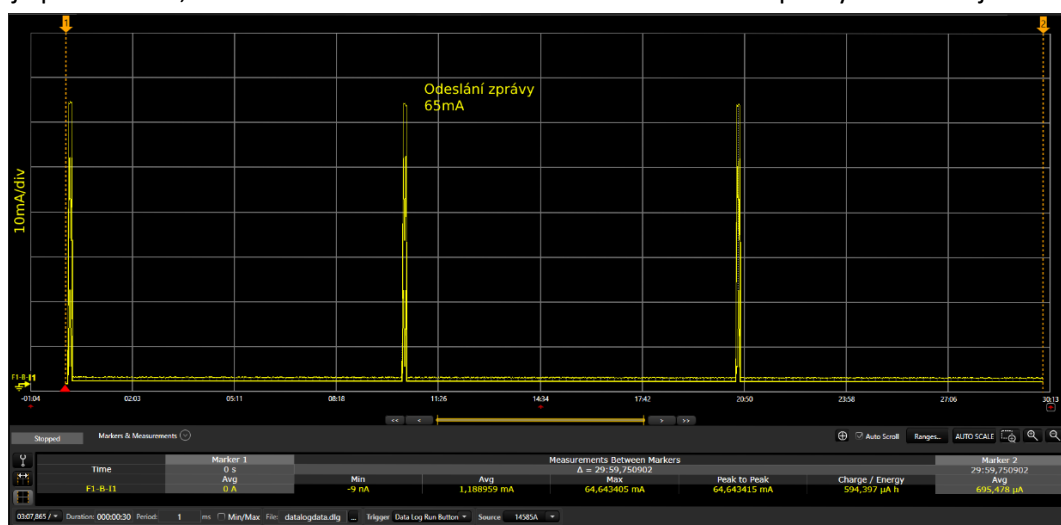
Obrázek 18 - Sestavený IoT senzor umístěný v krabičce

Krabička byla vytvořena na 3D tiskárně Prusa v prostorech Vysoké Školy Báňské, podle návrhu, který jsem vytvořil. Anténa je vyvedena skrze SMA konektor určený na montáž do panelu, pro termočlánek se ve spodní části nachází otvor o průměru bužírky na izolaci, tím je zajištěna určitá těsnost. Víčko krabičky je pak připevněno čtveřicí vrutů, pro které je krabička v rozích vyztužena.

2.4 Měření na hotovém senzoru

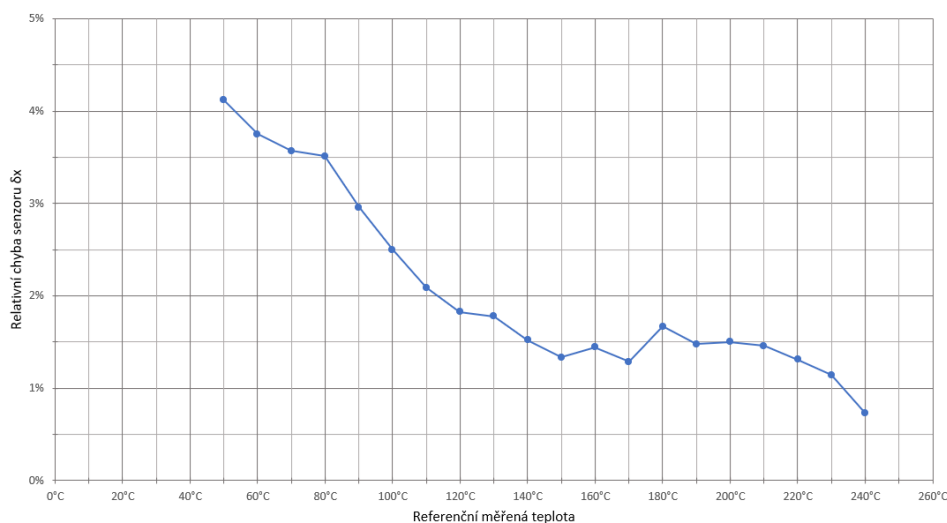
Na sestaveném senzoru byla provedena dvě měření, jedno za účelem zjištění jeho průměrné spotřeby a tím teoretické doby provozu na osazenou baterii, a druhé pro zjištění přesnosti měřené teploty.

V prvním případě byl pro určení spotřeby modulu použit DC analyzátor napájení Keysight N6705C, s nastaveným výstupním napětím 3,6V. Měření probíhalo v časovém horizontu 30 minut, během kterých došlo 3x k odeslání dat o teplotě. Výsledky měření lze vidět na Obrázek 19. Při odeslání zprávy roste odběr špičkově až na 65 mA, v režimu spánku pak pouze 0,6 mA. Průměrná spotřeba v časovém úseku je pak rovna 1,2 mA. Teoretická životnost na osazenou baterii kapacity 8500mAh je 295 dní.



Obrázek 19 - Měření spotřeby senzoru

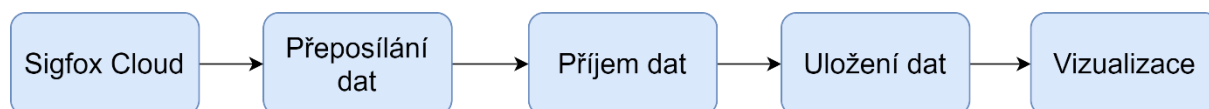
Při měření přesnosti jsem jako referenční čidlo využil multimetr UIN-T UT139C spolu s termočlánekem typu K který je k němu standartně dodáván. Měření probíhalo v peci s počáteční teplotou 50°C která byla postupně zvyšována v krocích po 10°C až na teplotu 240°C. Termočlánky byly umístěny v těsné blízkosti a po dosažení teploty byl vždy zanechán krátký časový úsek pro její ustálení. Výsledky měření lze vidět na Obrázek 20, kde je vynesena relativní chyba vůči referenčnímu čidlu, tabulka naměřených hodnot je k dispozici v příloze Příloha H -.



Obrázek 20 - Relativní chyba senzoru

3 Návrh softwarové aplikace pro zpracování a vizualizaci naměřených dat

V této části práce se budu zabývat přeposíláním dat, jejich zpětnou rekonstrukcí do původní podoby, dále pak vhodným uložením do databáze a vizualizací.



Obrázek 21 - Blokový digram zpracování dat

3.1 Přeposlání dat ze Sigfox cloudu

Zpráva odeslaná senzorem skrze RF komunikaci je zachycena všemi přijímači v dosahu a přeposlána do backend serverů. Ty provedou její zpracování a uloží jí na cloud, spolu s údaji o kvalitě přenosu, času obdržení, ID vysílače a přibližné poloze. Přístup k uloženým datům je umožněn skrze webové rozhraní na adrese <https://backend.sigfox.com/>. Zde je nutné se přihlásit k účtu ke kterému je zařízení registrováno a následně vybrat konkrétní zařízení ze seznamu, u kterého lze následně procházet veškerá zmíněna data, zobrazená v přehledné tabulce.

Pro přeposlání dat do vlastních serverů je k dispozici možnost definovat custom callback. Jedná se o uživatelsky definovaný http požadavek, který předá přijatou zprávu spolu s dalšími volitelnými informacemi na adresu zadaného serveru. Pro správný přenos dat je nutné nastavit vhodně jeho parametry, které rozeberu níže. Nastavení callbacku použitého při realizaci této práce je na Obrázek 22.

Device type LPWANcz_DevKit_1 - Callback edition

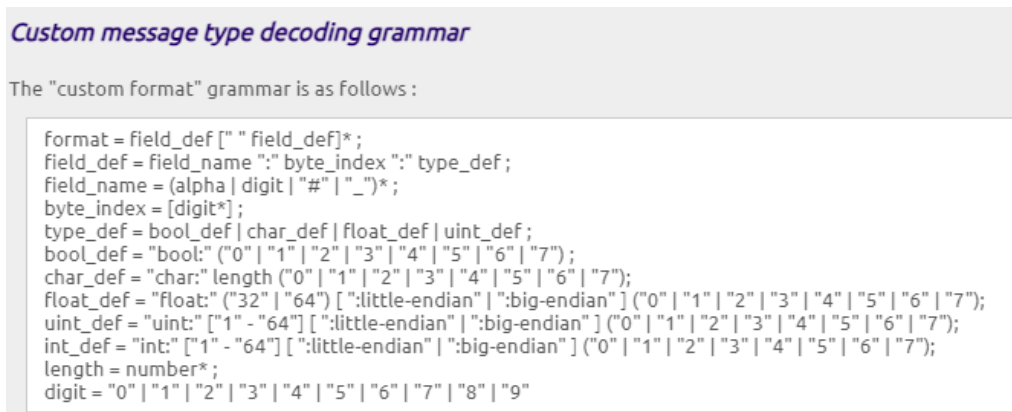
The screenshot shows the 'Callbacks' configuration window for a device. It includes fields for 'Type' (set to DATA), 'Channel' (set to URL), and 'Custom payload config'. Below these, there is a 'URL syntax' section with a pattern: `http://host/path?id={device}&time={time}&key1={var1}&key2={var2}...`. The 'Url pattern' field is filled with `http://www.sigfox.cz/insert.php?id={device}&time={time}&payload={data}`. The 'Use HTTP Method' is set to 'GET'. There is a checkbox for 'Send SNI' which is unchecked. The 'Headers' section shows 'Authorization' as the header name and a token as the value. At the bottom, there are 'Ok' and 'Cancel' buttons.

Obrázek 22 - Aktuální nastavení callbacku pro přeposílání dat na server

Type udává, jaký druh dat bude předáván, lze volit mezi DATA, SERVICE, ERROR, přičemž DATA slouží pro předání informací obdržených ze zařízení. SERVICE pro předání údajů o službě. ERROR předá chybové stavy. Dále lze zvolit směr komunikace, buďto UPLINK pro přenos dat odeslaných zařízením nebo BDIR pro obousměrnou komunikaci.

Channel umožňuje zvolit jakým způsobem budou data předána, lze volit mezi URL, BATCH URL, EMAIL. URL zašle data jako request na zadanou adresu. BATCH URL je vhodný, pokud je nutno odeslat data z většího množství zařízení, místo vytvoření jednotlivých URL requestu pro každé z nich, jsou data odesílána cyklicky každou vteřinu v rámci pouze jednoho požadavku, to snižuje špičkové zatížení serverů. EMAIL odešle data jako zprávu na zvolenou emailovou adresu.

Custom payload config slouží pro dekódování, parsování přenesených dat ještě před jejich odesláním. Pro správné nastavení této části je k dispozici tabulka obsahující příklady jednotlivých převodů.



Obrázek 23 - Tabulka pro nastavení custom payload config

URL pattern Zde se nachází adresa samotného serveru, který je volán, následována zvolenými proměnnými, které mají být předány.

Headers Umožňuje přidat http hlavičku. Tuto možnost jsem využil pro autorizaci, přičemž do hlavičky je vložen generovaný klíč, který je později kontrolován.

3.2 Příjem dat ze Sigfox cloudu

Jak bylo zmíněno v předchozí kapitole, data jsou z cloudu získávána za pomoci http požadavku get na zadanou URL serveru. Jeho realizace bude popsána v této části, spolu s PHP skriptem, kterému jsou data předávána.

Pro vytvoření webového serveru jsem využil software Apache. Jedná se o volně dostupný open source software, díky těmto vlastnostem se stává častou volbou a v dnešní době zajišťuje chod mnoha webových stránek. Jádro programu lze rozšířit o moduly, které mu přidávají nové funkcionality, toho jsem využil a jádro rozšířil o modul PHP který umožňuje spouštět skripty v tomto jazyce.

PHP je skriptovací jazyk, pomocí kterého lze vytvářet dynamické webové stránky. V práci je použit pro dekódování zprávy, kterou server obdrží a následný zápis proměnných do databáze, pro usnadnění je k zápisu využito knihovny mysqli. Soubor obsahující tento skript je uložen přímo na webovém serveru pod názvem insert.php a je obsažen v URL adrese na kterou jsou data zasílána Obrázek 22. Tím že zajištěno že webový server pouze obdržená data předá skriptu.

Ten ve své úvodní části zkontroluje, zda http hlavička obsahuje správný autorizační klíč, pokud ne, vrací error 403 tedy přístup odmítnut. Pokud autorizace proběhne v pořádku pokračuje skript rozdělením zprávy na jednotlivé proměnné, které poté převede na původní tvar. U teploty ještě proveden kontrolu proměnné sign a dle jejího stavu buďto vynásobí hodnotu -1 nebo jí ponechá. Skript pokračuje vytvoření instance třídy mysqli do které jsou zapsány přihlašovací údaje spolu s adresou a názvem databáze. Následně je proveden zápis proměnných do určených pozic. Při zápisu jsou vstupy do databáze ošetřeny proti SQL injection útoku a je zde zavedena kontrola správného zápisu. V případě chyby při zápisu je vrácen kód 500 tedy chyba serveru.

3.3 Uložení dat

Pro dlouhodobou archivaci dat jsem využil MariaDB databáze. Jedná se o vývojovou větev databáze MySQL se kterou je zpětně kompatibilní. Její nastavení a definice byly provedeny zasláním série příkazů v jazyce SQL. Struktura navrhnuté databáze je tvořena tabulkou, sestávající ze 4 sloupců v pořadí čas, teplota, ID zařízení a napětí baterie. Každý sloupec nese název proměnné, kterou uchovává a je nastaven na určitý formát dat.

```
mysql> select * from teplota
-> ;
```

time	temperature	device_id	voltage
2021-03-16 00:21:39	22.81	37BE5E	3.4
2021-03-16 00:31:59	22.56	37BE5E	3.4
2021-03-16 00:42:19	22.21	37BE5E	3.4
2021-03-16 00:52:39	22.27	37BE5E	3.4
2021-03-16 01:02:59	21.9	37BE5E	3.4
2021-03-16 01:13:18	21.77	37BE5E	3.4
2021-03-16 01:23:38	21.51	37BE5E	3.4
2021-03-16 01:33:49	21.37	37BE5E	3.4
2021-03-16 01:44:09	21.09	37BE5E	3.4
2021-03-16 01:54:28	21.05	37BE5E	3.4
2021-03-16 02:04:47	20.95	37BE5E	3.4
2021-03-16 02:15:07	20.83	37BE5E	3.4
2021-03-16 02:25:26	20.66	37BE5E	3.4
2021-03-16 02:35:46	20.55	37BE5E	3.4
2021-03-16 02:46:05	20.45	37BE5E	3.4
2021-03-16 02:56:23	20.43	37BE5E	3.4
2021-03-16 03:06:43	20.29	37BE5E	3.4
2021-03-16 03:17:02	20.14	37BE5E	3.4
2021-03-16 03:27:21	20.24	37BE5E	3.4
2021-03-16 03:37:41	20.12	37BE5E	3.4
2021-03-16 03:47:59	20.12	37BE5E	3.4
2021-03-16 03:58:18	19.83	37BE5E	3.4
2021-03-16 04:08:37	19.78	37BE5E	3.4
2021-03-16 04:18:57	19.95	37BE5E	3.4

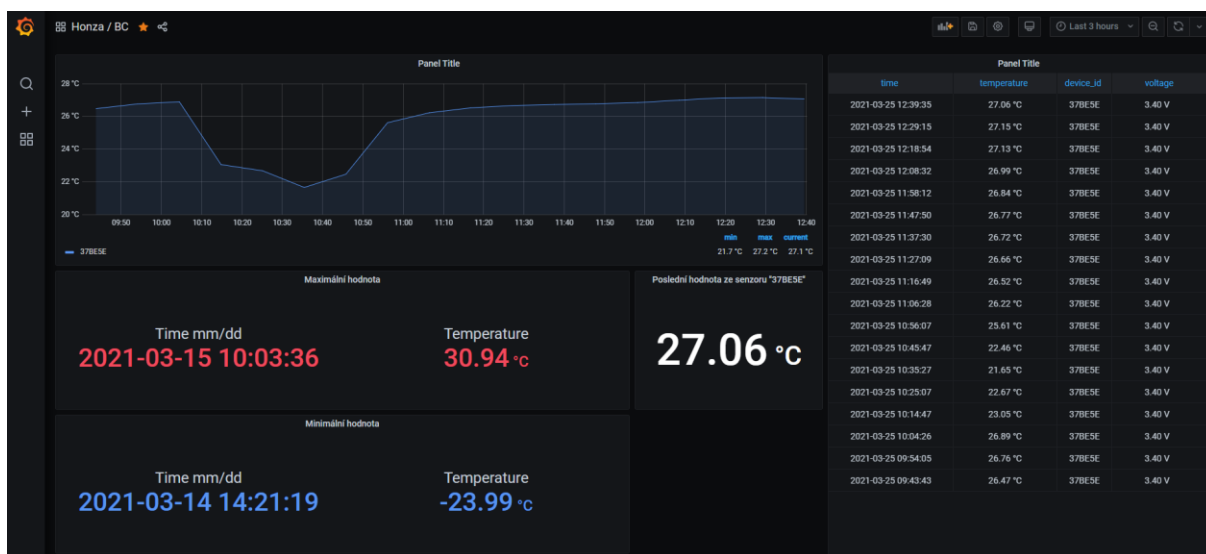
Obrázek 24 - Ukázka struktury databáze v příkazovém řádku

3.4 Vizualizace

Konečným článkem řetězce je vhodná vizualizace dat v časovém měřítku. K tomuto účel jsem využil software Grafana. Jedná se opět o open source volně dostupný software, který umožňuje vytvářet obrazovky označované jako dashboard které pak obsahují panely s proměnnými. Samotná vizualizace je pak dostupná skrze webovou aplikaci, pro svůj chod využívá Grafana rovněž webového serveru Apache popsaného v kapitole 3.2.

Po vytvoření dashboard je možné do její plochy vkládat jednotlivé panely. K dispozici je základní sada panelů, kterou lze rozšířit o komunitní panely. U každého panelu je nutné zvolit zdroj dat, v našem případě databázi vytvořenou v předešlé kapitole a v ní konkrétní tabulku. Poté je možné definovat barvy velikost způsob nebo jednotky ve kterých má být zvolená veličina zobrazena, možnosti jsou zde opravdu rozsáhlé.

Na Obrázek 25 lze vidět návrh vizualizace vytvořené pro tuto práci. Panel s grafem a tabulkou zobrazují hodnoty ve zvoleném časovém měřítku, zatímco maximální a minimální teplota je určována z celé databáze.



Obrázek 25 - Vizualizace dat ze senzoru v prostředí Grafana

4 Závěr

Cílem bakalářské práce bylo navrhnout a realizovat IoT senzor pro měření vysokých teplot na bázi termočládku, s využitím platformy Sigfox a následně navrhnout aplikaci pro zpracování a vizualizaci měřených hodnot.

Pro návrh bylo nutné se nejprve seznámit s platformami internetu věcí. Kromě využití Sigfox jsem dále vybral LoRa, IQRF a NB-IoT. Tyto platformy jsem nejprve teoreticky rozebral z hlediska bezdrátového nosiče který využívají, architektury sítě a komunikačního protokolu. Následně jsem od každé platformy vybral jeden dodávaný modul a popsal jeho periferie a použití.

Hardware senzoru má podobu jedné desky plošného spoje, na které je umístěn obvod pro měření teploty termočládkem spolu s lineárním regulátorem napětí. Komunikace s IoT platformou Sigfox je zajištěna modulem, který je k desce připojen pomocí konektoru. Řídícím prvkem celého senzoru je pak mikrokontroler, který využívá dvou sběrnic, I2C pro měřicí obvod a UART pro modul Sigfox, skrze které získává, upravuje a odesílá data. Deska plošného spoje je pak umístěna ve vytištěné krabici spolu s držákem baterie. Pro připojení externí antény je z krabice vyveden odpovídající konektor, pro vyvedení termočládku pak slouží otvor ve spodní části krabice.

Software pro mikrokontrolér byl navržen s důrazem na co nejnižší spotřebu. Sigfox umožňuje přenést pouze 140 zpráv denně, tedy průměrně jednu zprávu každých 10 minut a 18 vteřin, v mezičase je tedy mikrokontrolér, spolu s měřicím obvodem udržován v režimu spánku s nízkou spotřebou. Jeho probouzení a celý chod programu je řízen pomocí časovačů a přerušení. V této části práce jsem se potýkal s problémem náhodných chyb při komunikaci s modulem Sigfox, jejich příčinou byl nedostatečný čas pro ustálení taktu po probuzení mikrokontroléru. Tento čas byl upraven pro snížení spotřeby, ale byl jsem nucen jej vrátit do výchozích hodnot.

Pro vizualizaci a zpracování jsem využil sady open source technologií LAMP. Na serveru pracujícím s operačním systémem Linux je spuštěn webový server, který přijímá data z cloudu Sigfox prostřednictvím zpětného volání. Data jsou předána PHP skriptu, který je zpracuje a uloží do relační databáze. Vizualizace uložených dat probíhá s využitím softwaru Grafana, ve kterém jsem navrhnul panely pro výpis a vykreslení.

Senzor najde široké uplatnění v oblastech, kde je vhodné znát dlouhodobý vývoj teploty, ale jsou špatně přístupné nebo nebezpečné, jako jsou například haldy těžební hlušiny. Část vizualizace a zpracování je připravena na práci s více senzory, lze tedy sledovat i vzájemnou korelaci měřených dat. V budoucnu bych senzor doplnil solárním nabíjením, tím by byla doba jeho provozu násobně větší, oproti aktuálnímu stavu, kdy je nutné pravidelně měnit baterii.

Literatura

- [1] K. K. Okwara a C. A. U. Ugboaja. *The Internet of Things (Iot): A Scalable Approach to Connecting Everything* [online]. 2015, , 4 [cit. 2021-03-01]. ISSN 2319 – 1813, 2319 – 1805. Dostupné z: <http://www.theijes.com/papers/v4-i1/Version-1/C041109012.pdf>
- [2] GOKHALE, Pradyumna, Omkar BHAT a Sagar BHAT. *Introduction to IOT* [online]. 2018, , 5 [cit. 2021-03-01]. Dostupné z: https://www.researchgate.net/profile/Omkar-Bhat/publication/330114646_Introduction_to_IOT/links/5c2e31cf299bf12be3ab21eb/Introduction-to-IOT.pdf
- [3] ASHTON, Kevin. That 'Internet of Things' Thing. *RFID Journal* [online]. , 1 [cit. 2021-03-01]. Dostupné z: <http://www.itrco.jp/libraries/RFIDjournal-That%20Internet%20of%20Things%20Thing.pdf>
- [4] BOYES, Hugh, Bil HALLAQ, Joe CUNNINGHAM a Tim WATSON. *The industrial internet of things (IIoT): An analysis framework, Computers in Industry* [online]. 2018, , 101 [cit. 2021-03-01]. ISSN 0166-3615. Dostupné z: <https://www.sciencedirect.com/science/article/pii/S0166361517307285>
- [5] SLAMA, Dirk, Frank PUHLMANN, Jim MORRISH a Rishi BHATNAGAR. *Enterprise IoT* [online]. 1. USA: O'Reilly Media, 2015 [cit. 2021-03-01]. Dostupné z: https://books.google.cz/books?hl=cs&lr=&id=iqDUCgAAQBAJ&oi=fnd&pg=PR4&q=Enterprise+IoT&ots=vlv4ry6jFs&sig=kSUEkfQV-VmJYjeiGPOyExmLRe0&redir_esc=y#v=onepage&q=Enterprise%20IoT&f=false
- [6] *IQRF: Quick Start Guide* [online]. In: . 2020 [cit. 2021-03-01]. Dostupné z: https://static.iqrf.org/Quick_Start_Guide_IQRF_200918.pdf
- [7] *IQRF* [online]. [cit. 2021-03-01]. Dostupné z: <https://www.iqrf.org/>
- [8] *IQRF alliance* [online]. [cit. 2021-03-01]. Dostupné z: <https://www.iqrfalliance.org/technology>
- [9] [online]. [cit. 2021-03-28]. Dostupné z: <https://eshop.iqrf.org/products/detail/TR-72DA>
- [10] *Sigfox: Technical Overview* [online]. In: . Březen 2017 [cit. 2021-03-01]. Dostupné z: <https://www.disk91.com/wp-content/uploads/2017/05/4967675830228422064.pdf>
- [11] *SigFox* [online]. [cit. 2021-03-01]. Dostupné z: https://www.sigfox.com/en/what-sigfox/technology#id_technology

- [12] *Modul SigFox Node* [online]. In: . [cit. 2021-03-01]. Dostupné z: <https://partners.sigfox.com/products/lpwan-sigfox-node>
- [13] [online]. [cit. 2021-03-28]. Dostupné z: <https://build.sigfox.com/study>
- [14] POLÁČEK, Ing. Tomáš. *IoT síť Sigfox: Výhody, využití a implementace specializované sítě pro Internet věcí* [online]. In: . [cit. 2021-03-01]. Dostupné z: http://robodoupe.cz/wp-content/uploads/2017/04/SimpleCell_Sigfox.pdf
- [15] [online]. [cit. 2021-03-28]. Dostupné z: <https://www.sghosly.com/p/lora-is-chirp-spread-spectrum.html>
- [16] [online]. [cit. 2021-03-28]. Dostupné z: <https://ww1.microchip.com/downloads/en/DeviceDoc/RN2483-Data-Sheet-DS50002346E.pdf>
- [17] *3GPP Low Power Wide Area Technologies* [online]. In: . [cit. 2021-03-01]. Dostupné z: <https://www.gsma.com/iot/wp-content/uploads/2016/10/3GPP-Low-Power-Wide-Area-Technologies-GSMA-White-Paper.pdf>
- [18] [online]. [cit. 2021-03-28]. Dostupné z: <https://www.iot-portal.cz/2016/04/30/narrowband-iot>
- [19] [online]. [cit. 2021-03-28]. Dostupné z: <https://www.4gitemall.com/quectel-bc95-g-lte-nb-iot-module.html>
- [20] [online]. [cit. 2021-03-28]. Dostupné z: <https://cdn.sos.sk/productdata/76/21/ab3b6a18/bc95-b20-bc95-b20hb-02-std.pdf>
- [21] <https://download.mikroe.com/documents/datasheets/Quectel-BC95-G-V1.2.pdf>
- [22] JOST, Danny. Fierce Electronics [online]. [cit. 2020-11-29]. Dostupné z: <https://www.fierceelectronics.com/sensors/what-a-temperature-sensor>
- [23] [online]. [cit. 2021-03-28]. Dostupné z: <https://en.globtek.com/datasheets/ER34615M.pdf>

Seznam příloh

Přílohy v IS EDISON:

Příloha A - Elektrické schéma senzoru

Příloha B - Návrh desky plošného spoje

Příloha C - Program mikrokontroleru

Příloha D - Návrh krabičky pro senzor

Příloha E - Návrh víčka krabičky pro senzor

Příloha F - PHP skript pro zpracování dat

Příloha G - Měření spotřeby senzoru

Příloha H - Měření přesnosti senzoru